**Darlington, M., Thangarajah, U. and Ball, A. (2012) RAIDmap Application User Guide. Other. University of Bath, Bath, UK. (Submitted)**

**Link to official URL** (if available):

**Opus: University of Bath Online Publication Store**

http://opus.bath.ac.uk/

This version is made available in accordance with publisher policies. Please cite only the published version using the reference above.

See http://opus.bath.ac.uk/ for usage policies.

Please scroll down to view the document.

# RAIDMAP APPLICATION USER GUIDE

**MANSUR DARLINGTON, UDAY THANGARAJAH & ALEX BALL**

redm5rep120511mjd10.pdf

ISSUE DATE: 31May 2012

*Catalogue Entry*

| | |
|---|---|
| Title | RAIDmap Application User Guide |
| Creator | Mansur Darlington |
| Contributor | Uday Thangarajah, Alex Ball |
| Subject | research data management; data documentation |
| Description | This guide describes the functionality of the RAIDmap Associative Tool and explains how it may be used to document a research activity, the data records it produces as a result, and the associations between such records. |
| Publisher | University of Bath |
| Date | 12 May 2012 (creation) |
| Version | 1.0 |
| Type | Text |
| Format | Portable Document Format version 1.4 |
| Resource Identifier | redm5rep120511mjd10.pdf |
| Language | English |
| Rights | © 2012 University of Bath |

*Citation Guidelines*

Mansur Darlington, Uday Thangarajah & Alex Ball. (2012). *RAIDmap Application User Guide* (version 1.0). REDm-MED Project Document redm5rep120511mjd10. Bath, UK: University of Bath.

## CONTENTS

**1. INTRODUCTION ..... 4**

**2. APPLICATION LIMITATIONS ..... 5**

**3. THIS GUIDE ..... 5**

**4. INSTALLATION ..... 5**

**5. 'MAPPING' YOUR RESEARCH DATA RECORDS ..... 6**

**5.1 Opening an existing project ..... 6**
**5.2 Creating a new project ..... 6**
**5.3 Creating a data case ..... 8**
**5.4 Populating a data case ..... 8**
*5.4.1 Creating a data record map-object ..... 9*
*5.4.2 Recording a Relationship between data records ..... 13*
**5.5 Information about Nodes ..... 17**
**5.6 Exporting a RAIDmap record ..... 17**
**5.7 Importing a RAIDmap record ..... 18**
**5.8 Making copies and clones of research data map objects ..... 18**
**5.9 The Trash Bin ..... 18**
**5.10 The Inbox ..... 18**
**5.11 User Management ..... 18**

**6. REFERENCES ..... 19**

**APPENDIX A. A GLOSSARY OF TERMS ..... 20**

## 1. INTRODUCTION

The RAIDmap application allows users to record the existence and development of and the relationships between **Research Data Records** created during a research project. The application uses an informal representation of the UML-based Research Activity Information Development (RAID) model created during the JISC-funded ERIM and REDm-MED research projects, these projects focusing on improving the management of engineering research data. In this model are identified a set of different sorts of data records commonly encountered in research work, and a set of 'information development' activities or processes by which data records come into being as research is carried out.

RAIDmap allows during-creation and post-creation mapping of data records, and provides a visual representation of the data development context to help users make sense of the research data records and data sets associated with a particular research project or **Research Activity**. The purpose of this is to assist the researcher or research team in during-project data management activities, and to support post-project re-use of research data by those who have had no prior familiarity with the research project and the research data that have been gathered and generated. RAIDmap allows the simultaneous mapping of both the research data proper and the documentation which provides the basis for its interpretation.

RAIDmap works in two modes. It can be used entirely manually to select and associate research data records (RDRs) as a post-creation data management exercise; or it can be used in conjunction with 'RAIDwatch'.

RAIDwatch will alert the user when new files are created on the machine – files which may require 'mapping' – or when an attempt is made to delete a file which is part of a **RAIDmap Record**. At the same time RAIDwatch invokes the harvesting and recording of metadata when a file is created or modified. (The potential exists for RAIDwatch to be developed so that these and other similar 'housekeeping' functions can be fully integrated with RAIDmap.)

Central to the use of RAIDmap is the **RAID Diagram**, which illustrates visually the data records in the 'data case' and their relationships, associations and time line. A RAID diagram is developed by the user using a graphical interface.

The 'mapping' of research data records relies on the idea that each data record in the real world is represented by a data record node on the RAID diagram. A complex description, provided by a **metadata** record, is associated with each node on the RAID diagram, which describes the analogous data file in the real world. For clarity the terms map-object and real-object will be used to distinguish, where confusion might otherwise occur, between RAIDmap representations and the things in the real world that they represent.

The RAIDmap application is built using the Compendium information-mapping software in combination with the National Library of New Zealand's Metadata Extractor Tool.

This document is a deliverable (part of Deliverable 5. pt 4) of the REDm-MED Project funded by JISC as part of the second phase of the MRD Programme, a wider

exploration into managing research data resulting in the provision of policies, methods, procedures and tools in support of such management.

## 2. APPLICATION LIMITATIONS

RAIDmap v1.0 is an early prototype, created to demonstrate the potential of a visual mapping approach to managing research data records. Only the most basic mapping functions of the RAID Model are supported by RAIDmap v1.0. To encourage development of the application to implement the full power of the **RAID Modelling** approach and to maximize automation of data collection and, indeed, to encourage development of the model itself, the code is available from SourceForge under an open-source licence.

It is necessary for RAIDmap to 'recognize' a file type for the file-contained metadata to be harvested most effectively. Currently only the following file types are recognized by the metadata extractor: ARC, BMP, MS Excel, FLAC, GIF, HTML, JPG, MP3, OpenOffice, PDF, MS Powerpoint, Wave, MS Word, MS Works, XML. RAIDmap deals with other files by using a default harvester.

Details of how to extend the file types that are recognized are included in the *RAIDmap Application Developer Guide* (Ball, Thangarajah, & Darlington, 2012).

## 3. THIS GUIDE

The *RAIDmap Application User Guide* introduces the user to the RAIDmap functions currently implemented through the Compendium interface. Most of the RAIDmap functions, and usage of the user interface, are provided by Compendium native functionality. Access to the comprehensive Compendium's extensive help facility is available by selecting *Help* from the main menu; users are encouraged to use the two guides in conjunction with one another.

Where terms are used within the guide which have a special meaning in relation to the RAID Model, they are highlighted in bold at first use; and are defined in the glossary to be found in Appendix A.

## 4. INSTALLATION

Installation packages are available from SourceForge for 32-bit and 64-bit versions of Windows. Once you have downloaded the installation package, double-click on the package appropriate to your machine for installation to begin.

If installation is not initiated by double-clicking, the package can be installed from the operating system command line. The correct syntax is, as appropriate to your machine:

```
java -jar raidmap-install-32bit-windows.jar

java -jar raidmap-install-64bit-windows.jar
```

RAIDmap is an implementation of the Compendium software. In addition, RAIDmap utilizes two stand-alone software components: the Metadata Extractor Tool (for semi-automation of metadata extraction from data files) and the 'RAIDwatch' software. The

metadata extractor tool is integrated in the RAIDmap package and works in the background

RAIDwatch is a TSR (Terminate and Stay Resident, or daemon) application that alerts RAIDmap and the user when a new file is created modified or deleted on the host machine. The RAIDwatch programme must be downloaded and installed separately, and then launched automatically when the host machine is booted.

## 5. 'MAPPING' YOUR RESEARCH DATA RECORDS

Open RAIDmap by using the desktop icon created during installation or locating the program in the *Start – All Programs* menu.

By default RAIDmap opens with a blank working window.

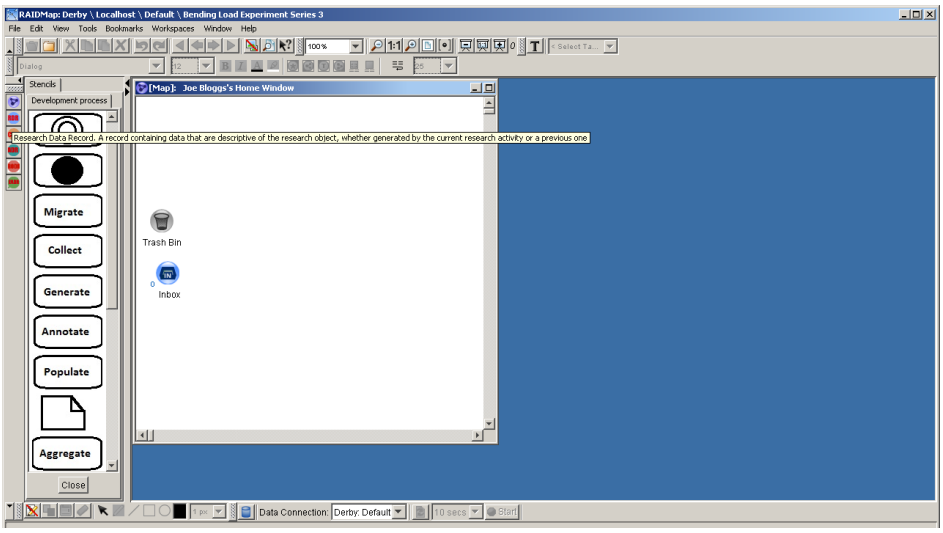
### *5.1 Opening an existing project*

Step 1: From the main menu select from the main menu *File – Open*.

Step 2: From the Project Log-in dialogue box select the preferred project and provide the user name and password.

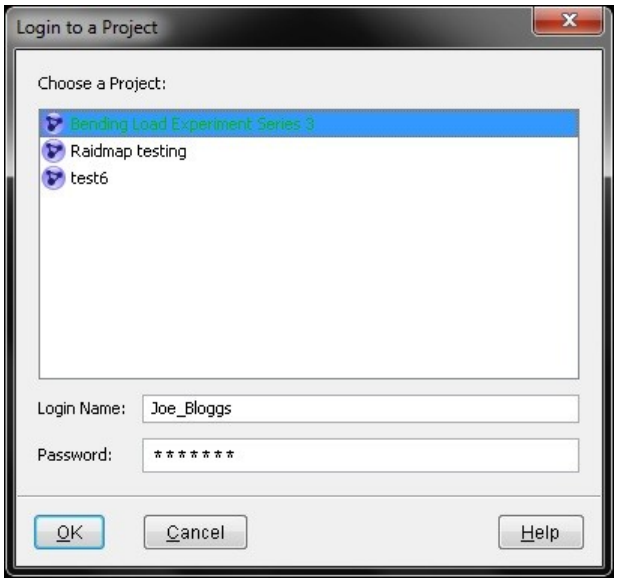
### *5.2 Creating a new project*

Step 1: If a project is already open; from the main menu select *File – Close,* else select *File – New*.


RAIDMap: Derby \ Localhost \ Default \ Bending Load Experiment Series 3
File Edit View Tools Bookmarks Workspaces Window Help
Stencils
Development process
[Map]: Joe Bloggs's Home Window
Research Data Record. A record containing data that are descriptive of the research object, whether generated by the current research activity or a previous one
Migrate
Collect
Generate
Annotate
Populate
Aggregate
Close
Trash Bin
Inbox
Data Connection: Derby: Default

Step 2: Enter project details, log-in name and password in the Create a New Project dialogue box, and click on *Create*.

Step 3: Select the newly created project from the Log-in to a Project dialogue box and click *OK*. A new working window will be displayed.




**Note:** Currently this log-in name and password is used only by the RAIDmap application.

The 'RAIDmap Project' is the over-arching research activity. For simple research projects it may be appropriate to use the name of the research project itself as the RAIDmap Project Name. For complex projects it may be desirable to work at some smaller sub-division of a research project, such as by work-package or task. The name of the open project is given in the application main window header bar.

**Note:** By default in the new working window is displayed a number of Compendium-related nodes: Trash Bin and Inbox.

The use of the Trash Bin and the Inbox are described in Sections 5.9 and 5.10 respectively.

### *5.3 Creating a data case*

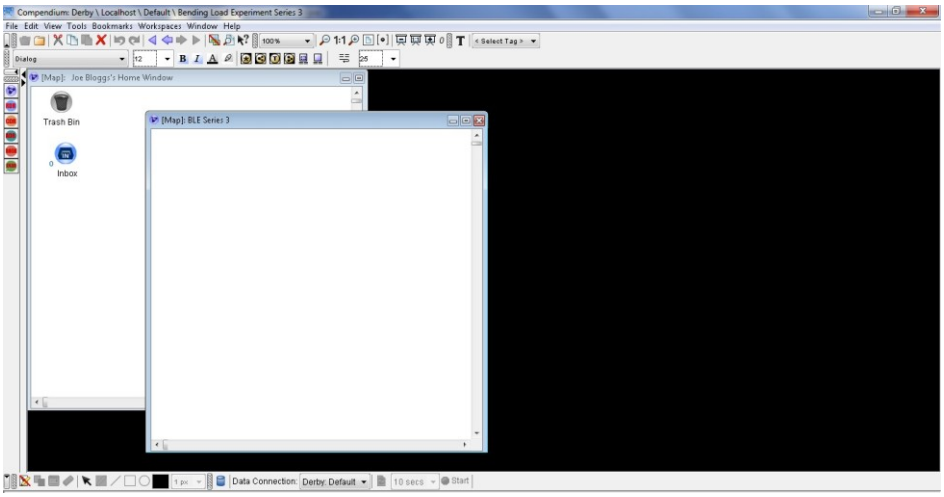
RAIDmap uses the idea that research work is carried out at different levels of organization. Often research is carried out in a single project. Where the project is small and simple, it may be appropriate for all the research data records and their contextualizing data records to be organized in one 'container', for which the term used by RAIDmap is the **Data Case**. If the Project is a simple one, the top-level, Project, window may be used for mapping the data case, that is to say, all the data records will appear in the Project Window. For other, more complicated, research work it may be appropriate to consider the research as being carried out in sub-activities and for the data records within these sub-activities to be organized into separate data cases. Such sub-activities might be organized at the level of the work-package, task or a single experiment. In this second case a set of data cases would be created in the top-level Project Window, the RAIDmap diagram for each data case being created in a separate window.

The user must elect at which level of research to organize the data and to create an appropriate number of data cases, each one of which should be named accordingly. Organizational decisions made now may be changed later; the hierarchical organisation of the RAIDmap data cases and data records they contain may be rearranged at any time.

Step 1: To create a RAIDmap data case, select the *Datacase* icon from the left-hand tool bar and drag it into the working window.

Step 2: Enter the name you wish to use for the data case in the box provided.

Step 3: Double-click on the data case to open the empty data case window.

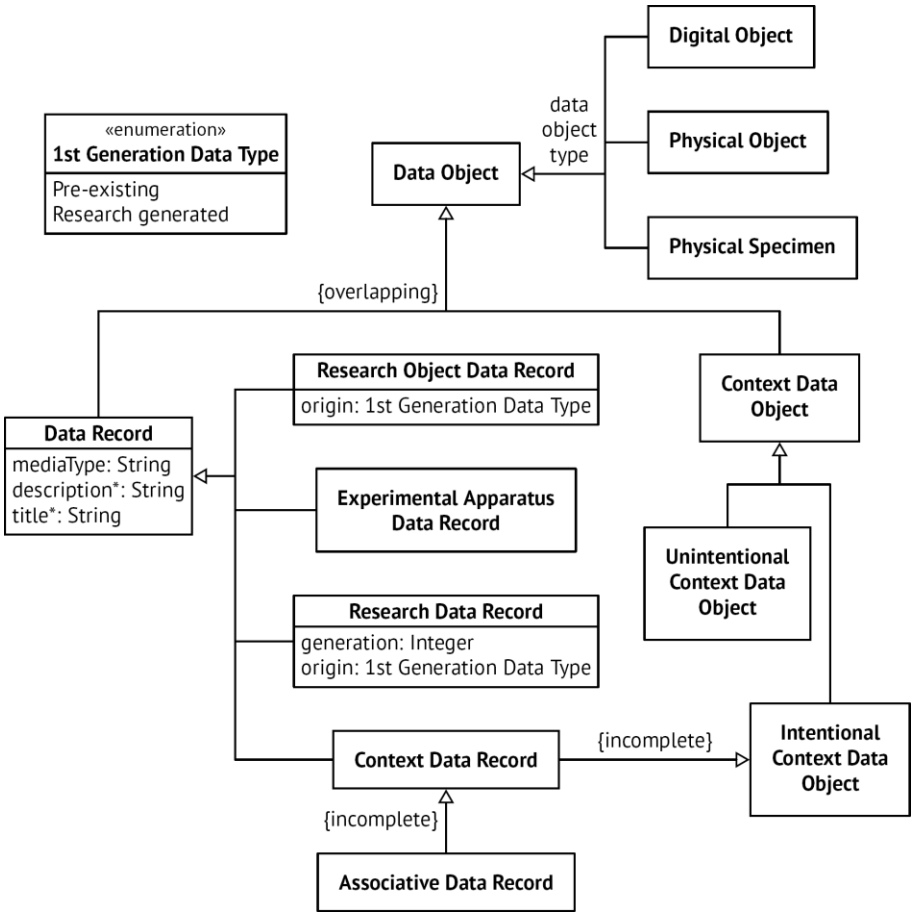



### *5.4 Populating a data case*

Once a data case has been created, it is necessary to add data records to the data case.

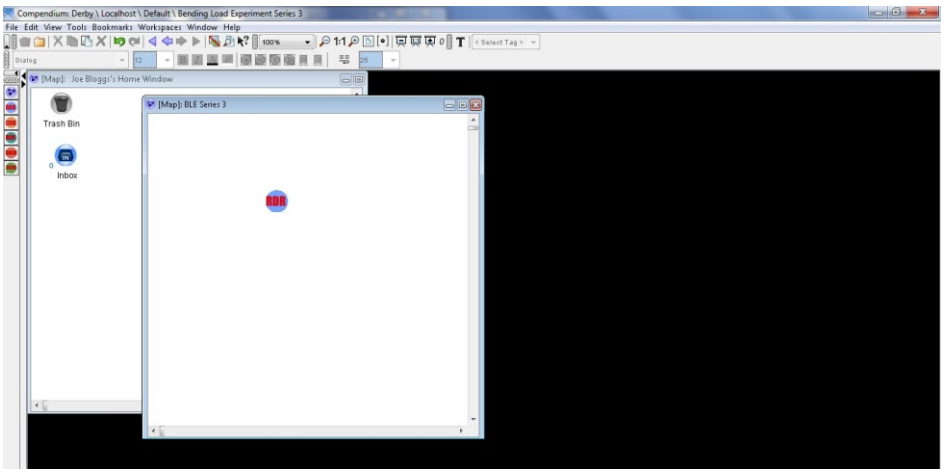
There are five types of data record identified and defined in the RAID model, as shown below; see **Appendix A** for definitions of the different types of **data record**. Records in RAIDmap are further classified as being either a **digital object**, a **physical object** or a physical specimen. The distinction is made in the use of the last two between *physical objects* (e.g. log books, printed matter, hand-written matter) – these being directly

analogous to digital objects in the form of data records – and *physical specimens,* such as experiment artefacts, which may contain or constitute data. The reason for the distinction in the RAIDmap application is that physical data records share many of the properties of digital records, physical specimens rather fewer.

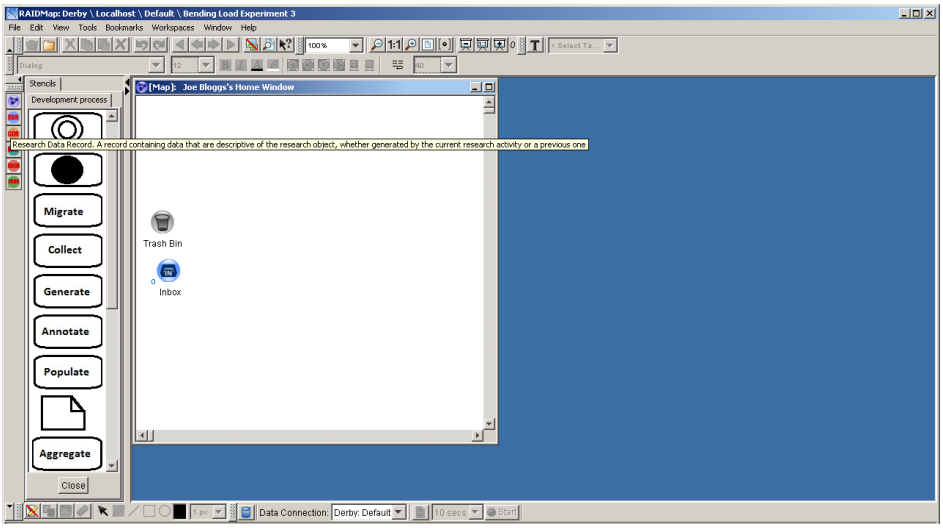



### *5.4.1 Creating a data record map-object*

Step 1: From the bank of available data record icons on the left-hand side of the screen select and drag an icon of the record type appropriate to the data record in the real world that you wish to map.



**Note:** Mousing over the RAIDmap icon identifies the data record type that it represents and provides a definition.


RAIDMap: Derby \ Localhost \ Default \ Bending Load Experiment 3
[Map]: Joe Bloggs's Home Window
Research Data Record. A record containing data that are descriptive of the research object, whether generated by the current research activity or a previous one
Trash Bin
Inbox
Migrate
Collect
Generate
Annotate
Populate
Aggregate
Close
Data Connection: Derby: Default

Step 2: Enter the name you wish to use for the data record – you could use the filename, title or a short description – then press Return to open the properties/contents box for the record.

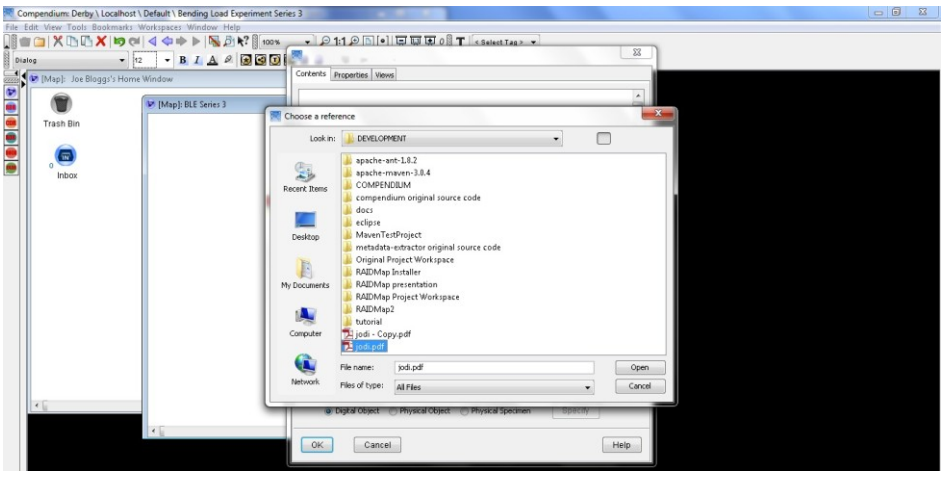
**Note:** If you leave the label blank, one will be generated for you from the filename of the real-object, if applicable.

**Note:** You can also open the properties/contents box by double-clicking on the record icon, or by right-clicking on it and selecting 'Contents' from the context menu.

**Note:** The properties/contents box presents information about the RAIDmap entity (represented by the node) and allows information to be displayed that has been captured and stored about the real-object that the node represents.

Step 3: Select from the check-box list: 'digital object', 'physical object', 'physical specimen' as best describes the real-object.

Step 4a: To associate a digital real-object with the map-object click on the *Browse* button and locate the digital file in your file space.

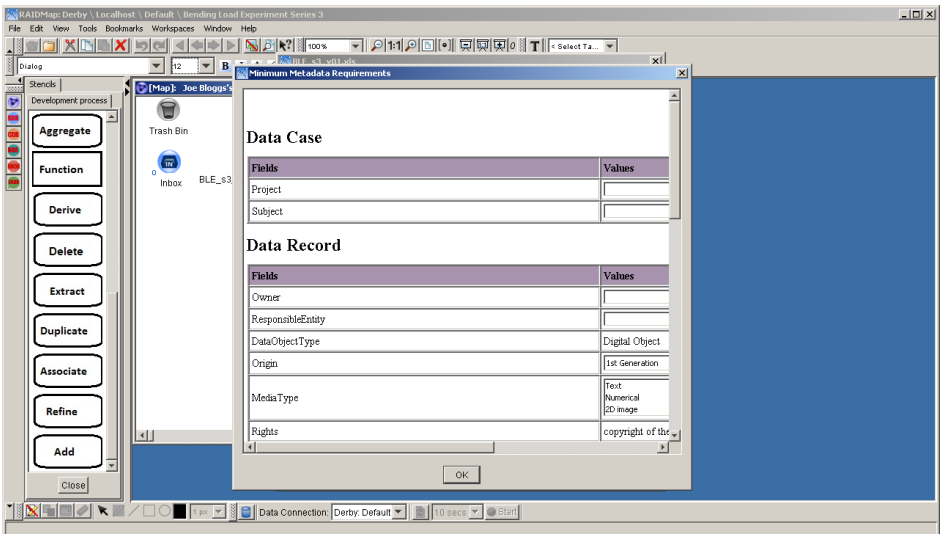


In the dialogue box, 'opening' a file will in fact associate it with the map-object, invoke the metadata harvester (this may take a while), and open the metadata window. The harvested metadata will appear in the window.

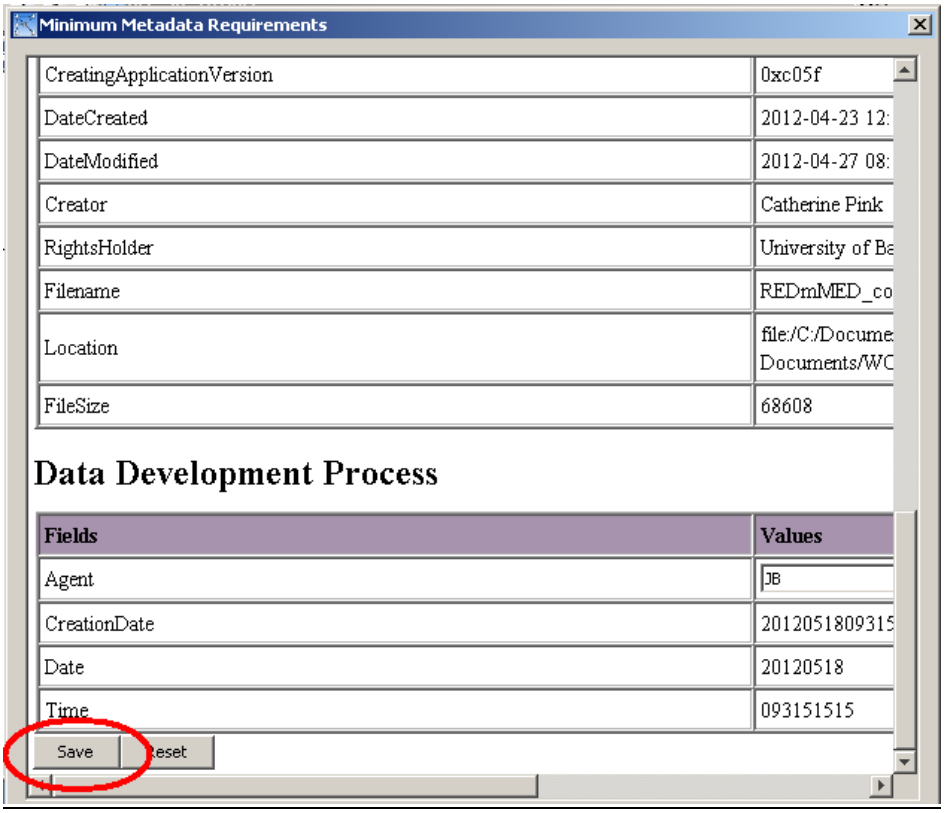
**Note:** Currently RAIDmap will handle only the file types listed above. Extending this range will rely on further development.

Step 4b: To associate a physical object or physical specimen with the map-object click on the *Specify* button.

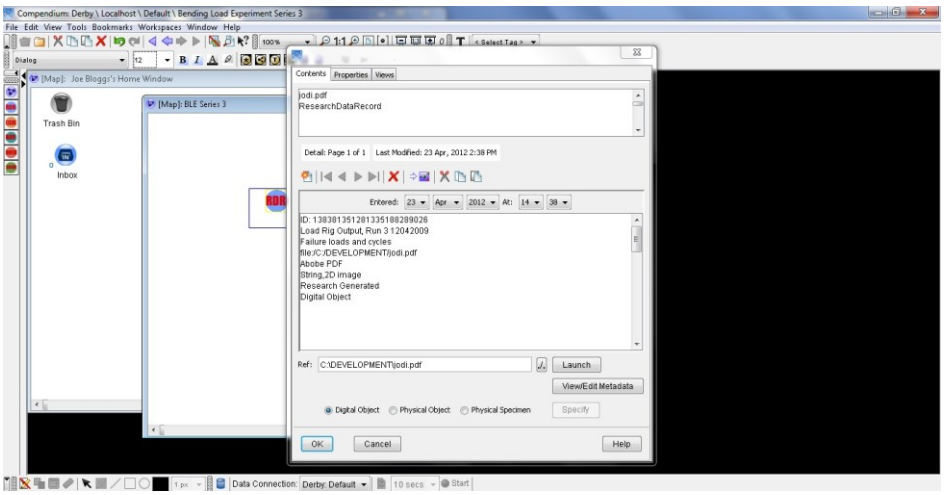
In each case, complete the metadata description in the Minimum Metadata Dialogue Box by providing the missing information. The extent of automatic completion will depend on the amount of metadata that the harvester has been able to collect. For digital objects this will depend on the file type and the metadata entries that have been made in the original file. For physical objects and specimens most of the metadata entries will have to be entered by the user.




When all the metadata have been entered scroll down to the bottom of the window and click the *Save* button. DO NOT click the *OK* button until **after** you have saved the metadata otherwise the user-provided entries will be lost.



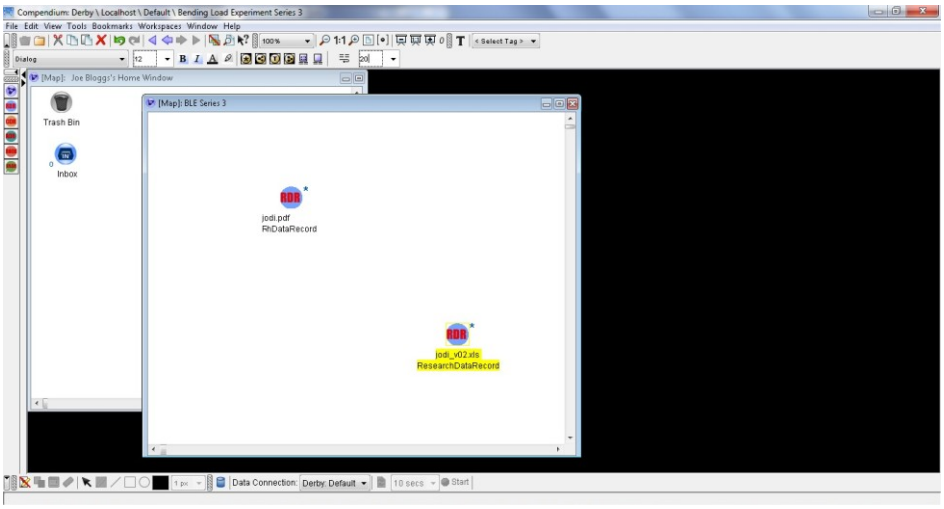

**Notes:** It is necessary to fill in all the blank metadata entries. If you attempt to save the metadata form when there are one or more entries missing the system will prompt you, one entry at a time, to provide the missing information.



Once the metadata has been stored a * will appear upper right next to the object's node in the working window. Mousing over the * will provide key metadata including such things as the document title, the file name and the description. The amount of data displayed is customizable; to make a change to the length of the display shown go to *Tools – User Options*; select the *Map & Rollover tab* and change amend the number in the *Detail rollover length* box.

The above steps may be repeated for each data record that requires mapping into the data case.

**Note:** Because of the way that RAIDmap handles the metadata extraction, if a data record is open when the association is made between a map-object and a real-object, the data record will be locked, meaning that it cannot be saved or deleted until RAIDmap is closed.

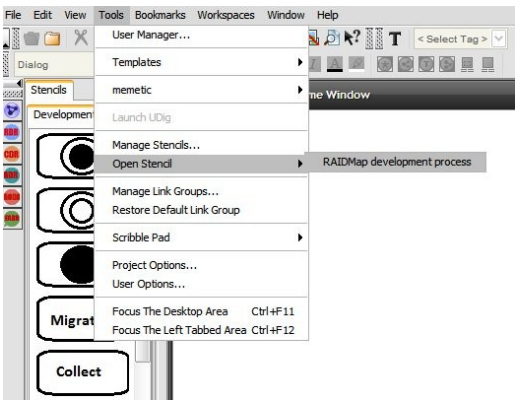



**Note:** The set of mandatory metadata collected by the application is defined in an XSLT configuration file. Details of how to locate and edit this file are given in the *Developer Guide*.

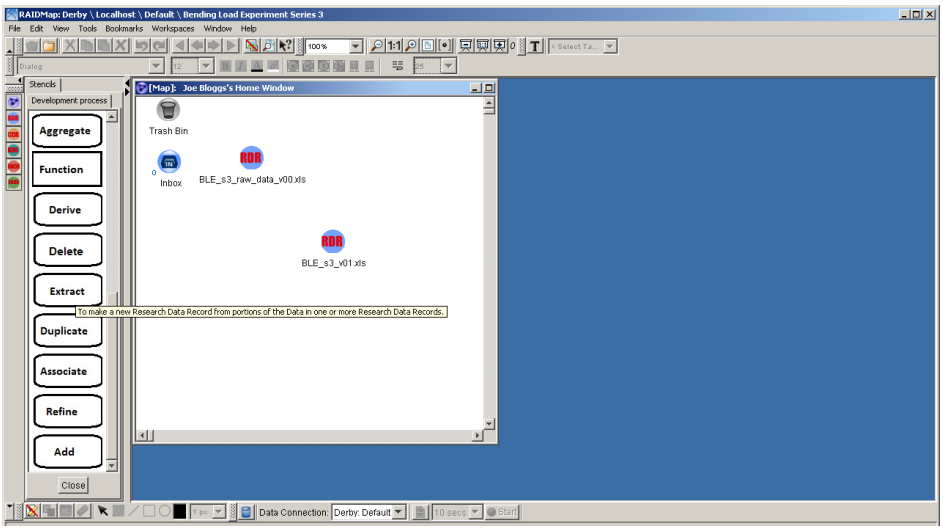
### *5.4.2 Recording a Relationship between data records*

Rarely will a data record exist in isolation; each will be connected with some other record in some way or have some reason for being created. Mapping the relationship between data records is achieved by indicating visually which of the Research Activity Information Development processes or activities (as described in the RAID model) led from one record to another. To record a relationship between data records RAIDmap uses a set of labelled icons which can be placed between map-objects and connected by arrows. The set of icons are found in the stencils palette displayed by default to the right of the RAIDmap DR icon menu.

Step 1: If the stencil palette is not open, from the main menu select *Tools – Open Stencil* and select 'RAIDmap development process' to open the stencil palette.

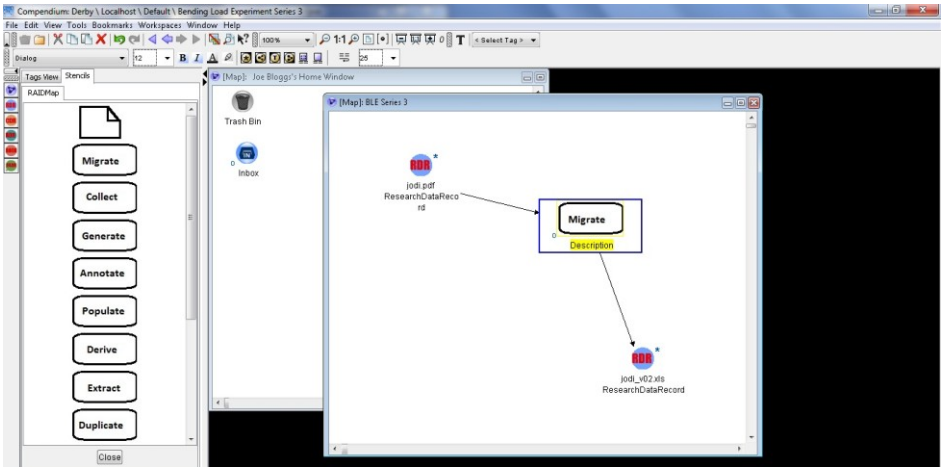



Hovering over a process icon in the process palette will provide a definition of that process.



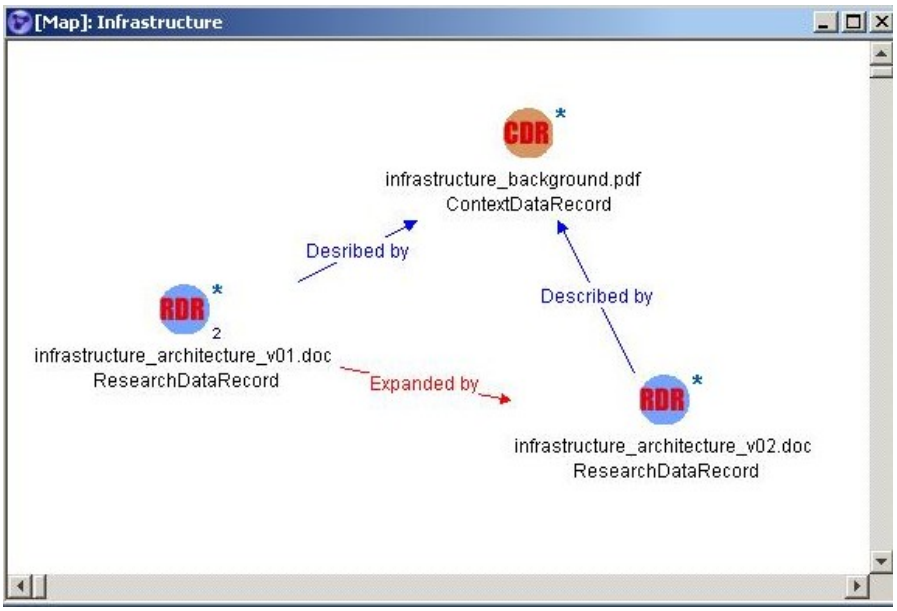

Step 2: Drag and drop the desired process/activity into the working window. Amend the description of the process with explanatory information as required.

Step 3: To link two data records through the selected process right click on the precursor data record node and drag an arrow to the target process node. Likewise create an arrow from the process node to the successor data record node.




Step 4: To link two data records or cases together without using an intermediate process create an arrow as in Step 3. The arrow can be given an explanatory label by clicking on the arrow to activate the description box. This can provide useful additional contextual information as shown in the examples in the following figure. Right-clicking on an arrow will bring up a context menu that allows changes to its direction, colour and so on.

**Note:** Arrow colours are currently associated with types of processes. Use of such semantics is entirely optional.




**Note 1:** For more detailed definitions of the RAID processes and a more formal description of the RAID Model, reference should be made to the on-line paper Visualizing Research Data Records for their Better Management. It has been found that the category in which a data record should be placed is sometimes unclear, the boundaries between classes being, in reality, somewhat fuzzy. So, don't agonize about choosing the right icon to represent the real-object; better to have a fuzzy record than none at all. The metadata content and the context in which the data record is found will help to clarify its rôle in the **RAID Diagram**.

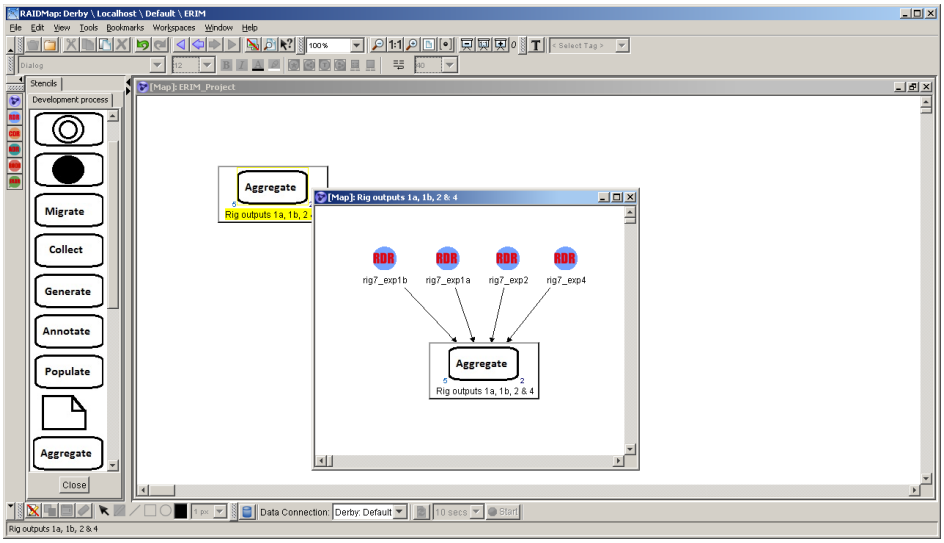
**Note 2:** *Aggregation* is the process of producing a larger set of data from a number of sources. Because of the potential of a large number of files being involved in an aggregation process, and the concomitant cluttering of the RAIDmap diagram, the following process should be followed to record aggregation:

Step 1: Drag and drop an aggregation icon from the Process Palette onto the work space and label as appropriate.

Step 2: Make a copy of the node by accessing the context menu using the right mouse button and selecting 'copy'.

Step 3: Double click on the aggregate node to open the associated window (named '[Map]: *Label*', where *Label* is the label you provided) and paste in the copied aggregation node.

Step 4: Drag into the '[Map]: *Label*' window an appropriate DR icon which represents one of the files from which the aggregation was created. Alternatively, where the data record is already recorded in the RAIDmap diagram, copy the node and paste it in. Within the '[Map]: *Label*' window, associate the input DR with the aggregation node by adding an arrow from one to the other. Complete the data record map-object creation process as shown in Section 5.4.1. Repeat the process as required to populate the aggregation window with the data records from which the aggregation data were drawn.




**Note 3:** The RAIDmap Process stencil palette has five icons additional to the RAIDmap processes, as follows:

*Start Event:* This is used to clarify where the start of a process occurs. For example, the first output of research data from a rig would be identified by the process 'generate'. The *generate* icon could be annotated with the *start* icon to indicate that this was the first in a chain of information development processes associated with the experiment.

*Intermediate Event:* This is used to indicate where a branch of research activity – but not the research activity as a whole – concludes; it is equivalent to a UML flow final node (⊗). A common use for this node is to indicate that the data record which precedes it has been **deleted.**

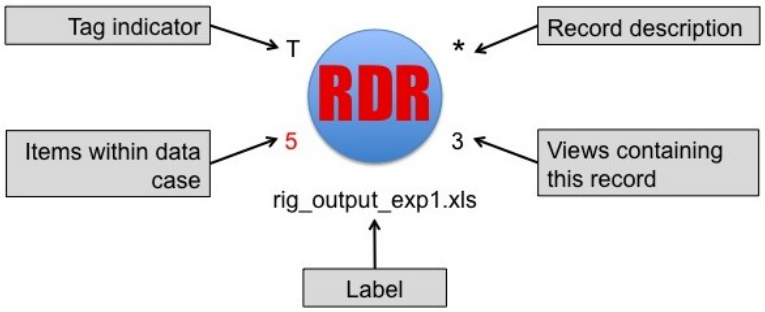
*End Event:* This is used to indicate the completion of a data case, and therefore the associated research activity. It follows the last data record to be created, or a final data development process.

*Function*: The Function icon provides the means for recording the details of functions applied to data in one record which have resulted in the data recorded in the successor data record. For example, when *migrating* from one data format to another, say from a word-processing format to PDF, it would be possible using 'Function' to record details of the converter used. Similarly, if a *refining* algorithm or method were applied to the contents of file, details of the algorithm or method could be recorded.

*Note:* The Note icon can be used to add general contextualizing information to facilitate interpretation of the RAIDmap diagram that cannot be recorded in any other manner.

### *5.5 Information about Nodes*

Compendium nodes (aka RAIDmap nodes) carry a number of information items in addition to the node type, as shown in the figure below.




The purpose of these information items and the node types to which they apply are as follows:

| Information item | Description | Data record node | Data case node |
|---|---|---|---|
| Label | Inherited from the metadata element file name for the data record with which the map-object is associated; the label can, however, be edited manually thereafter if required. Updating the label *does not* update the metadata record, however | √ | √ |
| Tag Indicator | User-defined tags associated with the data record | √ | √ |
| Record Description | Metadata digest | √ | n/a |
| Items within data case | The number of data records and/or sub-data cases that are mapped within the data case | n/a | √ |
| Views containing this record | The number of 'windows' in which instances of this same data record or case can be found | √ | √ |

All but the Label item may be switched on and off individually using the *Tools – Project Options – Node Extras* dialogue box or by using the Node Format properties toolbar. Mousing over the peripheral information elements gives the underlying detail information and clicking on the information item will open the dialogue box related to that information type.

### *5.6 Exporting a RAIDmap record*

Information about the data records in a data case is recorded through the means of the RAID diagram. It is possible, however, to export the record so that it can be viewed in

different ways. To export a RAIDmap Record, select from the main menu *File – Export* and then select the required output format. RAIDmap records exported in XML format may be imported, read and manipulated using a RAIDmap application (see next section). Export options also include two for reading using a web browser: Web Outline provides a manifest of the records in a data case, while Web Maps reproduces some of the look and feel of the RAIDmap Record in HTML using text and image maps.

### *5.7 Importing a RAIDmap record*

RAIDmap records which exist in XML form may be imported into a RAIDmap Project using the import function. To import a RAIDmap Record, select from the main menu *File – Import* and then select the format of the input file. A number of different formats are included such as XML and the facility to import the contents of an image folder.

### *5.8 Making copies and clones of research data map objects*

RAIDmap map-objects (including both DR and Process Nodes) may be both copied and cloned.

When an object is *copied* another instance of the same entity is created. This function allows a single entity to exist in different windows (i.e. data cases or aggregations) in a RAIDmap record: the Compendium term for this is 'transclusion'. To identify how may copies of the single entity exist a number is shown to the lower right of the data record or data case node.

When a map-object is *cloned*, an entity is generated that is identical to the original but for its unique, system-generated, identification number. The two clones may each be edited without affecting the other.

### *5.9 The Trash Bin*

The Trash Bin is one of two, non-deletable, work-top items which will appear in a project window. When a data record or data case is deleted or cut from a view it is placed in the Trash Bin. Until the item is purged from the bin it can be restored to the current working window or the window from which it was deleted. Access to the Trash Bin contents is by double-clicking on the bin icon in the Project main window.

### *5.10 The Inbox*

The Inbox is one of two, non-deletable, work-top items which appear in a project window. Essentially it may be used to provide a list of selected nodes in which is displayed node properties. Nodes may be sent to the Inbox of other users. The Inbox has no special function in RAIDmap; it is explained in detail in Compendium Help.

### *5.11 User Management*

A RAIDmap Project may be set up as a single-user or multi-user project either when the project is created or by using the *Tools – User Manager* dialogue box.

When a single-user project is set up it is possible for the user to elect to make a given project a default project.

When a multi-user project is set up, the name and log-in details of each user must be given by an administrator. The creator of a project is automatically made administrator by default. Other users may be made administrator as required. The rights of non-administrators is limited, for example they have no access to the *User Manager* facility nor to some project management tasks such as deleting projects and adding additional users.

## 6. REFERENCES

Ball, A., Thangarajah, U., & Darlington, M. (2012). *RAIDmap Application Developer Guide* (REDm-MED Project Document redm5rep120510ab10). Bath, UK: University of Bath.

## APPENDIX A. A GLOSSARY OF TERMS

The terms in this glossary are those to be found in this user guide together with some others chosen to provide context. They are a sub-set of those in the Terminology for Research Data Management which was an output of the ERIM Project and in which can be found accompanying explanatory text and sources.

The terms are not strictly in alphabetical order here but have been grouped loosely into logically associated clusters; the clusters are demarked by horizontal lines.

---

**RAID Diagram.** A 2-D graphical representation – based on the RAID Modelling method – of the Data Records within a Data Case and their associations and principal characteristics. The RAID diagram visualizes a part of the data contained within the RAID Record.

**RAID Modelling**. A method of modelling the development of Data during the Research Activity which identifies Data Records, their relations, associations and Metadata.

**RAID Record**. The underlying record of a Data Case containing information about Data Records constituting the case, their temporal ordering, relations and association and principal characteristics. Also a data file in which the record is contained identified as such by the RAIDmap application data type extension *.rmap*.

**Research Activity** The process through which research data and context data are accumulated and developed.

**Research Data Development Process** One of a set of processes that are commonly carried out during the research activity which changes or adds to the research data associated with a research activity or project.

---

**Addition (Add)** To supplement existing data at the data level, for example when inducting data into an existing file.

**Associate** To make explicit the relationship between items of data, data records or Data Cases (cf. Related).

Example: Two data records may be related by such things as file names, metadata, explicit reference in one to another, by an embedded link or by a separate document which cites the association (see: Associative Data Record).

**Aggregate** To combine similar data from different sources for the purpose of increasing sample size (cf. Augment).

**Annotate** To add 'information or additional marks formulated on a document for enhancing it with brief and useful explanations'.

Note: annotation may be made on the subject document itself or on a 'stand-off' document with the annotation linked to the annotated content.

**Augment** To add Research Data Records or Context Data Records (which see) to a Data Case.

**Collate** Give order to data assembled from different sources.

Note: This can occur at any organizational level, that is, at the data level, the data record level or the data case level.

**Collect** To acquire and bring together pre-existing data.

Note: This is concordant with the DCC's use of the term 'receive' in relation to pre-existing data collected from external sources.

**Data Cleaning** A special case of data Refinement (cf.), where the refined data are a normalized version of the source data; that is, with systematic errors corrected, calibration taken into account, invalid data removed, etc.

**Delete** To expunge or obliterate.

Pretty self-explanatory, but rather important in relation to data management.

**Derive** (cf. Refine) To create new data by applying logical inference, extrapolation, or similar algorithmic process to pre-existing data.

Note 1: Derivation constitutes creating a new description or representation of an analysis of the subject described by the existing data.

Examples: a histogram representing the frequencies of data occurrence; a narrative or commentary on a set of interviews.

**Duplicate** To make another, identical, copy of a file.

**Extract** To make a new research data record from portions of the data in one or more Research Data Records.

**Generate** To act on or interact with a research subject thereby creating research data.

**Migrate (Format Migration)** The transfer of digital information from one format to another (with the intention of preservation of the full information content) (DCC glossary)

**Populate** To enter data residing in a data record into an existing framework such as a pro-forma or a knowledge- or data-base.

**Refine** To re-express data in a different form or according to a different data model (cf. Derive).

Example of typical refinement functions: rounding, normalization, removing duplicates, stop-words, noise or outliers, simplification, etc.

**Transform** To derive or refine data in such a way that new or changed data results.

Note: this interpretation excludes Format Migration because in Migration - in contrast to derivation and refinement - there is expressly no intention to change the data.

---

**Data** Reinterpretable representations of information in a formalized manner suitable for communication, interpretation or processing.

Examples of data include a sequence of bits, a table of numbers, the characters on a page, the recording of sounds made by a person speaking, or a moon rock specimen. (OAIS Reference Model)

**Information** Any type of knowledge that can be exchanged. In an exchange, it is represented by data. An example is a string of bits (data) accompanied by a description of how to interpret a string of bits as numbers representing temperature observations measured in degrees Celsius (information).

Note 1: the 'accompanying description' referred to above could be substituted by knowledge appropriate to achieving the same interpretation.

Note 2: There is a logical tension apparent in the definitions of data and information in respect of the enterprise of making 'research data' re-usable. Research data are only re-usable if the information or knowledge is (made) available for correct interpretation to follow, in which case strictly it is 'information' not 'data'.

**Metadata** Data about data.

The term metadata is often used in a rather restricted way, for example limiting the scope to keywords, terms populating (xml, html) tags and so on. It is intended here to be interpreted to embrace ‘information about data’, and includes both data that is local to the data that it describes (i.e. in the same data object) or is held remotely in a separate file. All metadata ‘contextualizes’ the data to which it refers, some metadata is a means by which data is associated

---

**Record** Information in any medium, created, received and maintained as evidence of an activity.

**Recording (data)** Encoding the output from Data Generation in a carrier format (e.g. bit stream, written notes).

**Related** Two or more items of data, or data records which have an implicit or explicit connection. Explicit connections are made through Association (which see).

**Associative Data Record (ADR)** A record which documents the association between other data or data records. The data contained within an ADR is a special case of contextualizing data and the data record a special case of a CDR.

**Data Record (DR)** The data object which contains the data.

**Experimental Apparatus Data Record (EADR)** A digital object which is analogous to the physical experimental apparatus familiar in much laboratory-based research.

**Research Data Record (RDR)** A record containing research data, i.e. data that is descriptive of the research object.

**Context Data Record (CDR)** A record containing data explicitly intended to place in context other data or abstract aspects of the research subject or activity.

A special case of the CDR is the Associative Data Record.

**Research Object Data Record (RODR)** A data object which is itself the object of research interest or which together with other RODRs constitutes the object of research interest.

**Data Case** The set of data records associated with some discrete research activity (project, task, experiment, etc.).

---

**Data Object** Either a Physical Object or a Digital Object.

**Digital Object** An object composed of a set of bit sequences.

**First-Generation Data** The data resulting either from data collection or from data generation.

This term is intended to identify data which has not been the subject of derivation or refinement